

Please replace the paragraph beginning at page 1, line 16, with the following rewritten paragraph:

a¹
Existing communications systems have had a long time to establish security mechanisms
5 and to build up trust in them by their users. In the United States our conventional postal mail is a
good example. We deposit our posted letters into a receptacle which is often very physically
secure. Our letters are then picked up, sorted, transported, and ultimately delivered to a similar
receptacle for retrieval by their recipients. Between the receptacles of a sender and a receiver the
persons handling a letter are part of a single organization (at least intra-nationally) that is well
10 known to us and considered to be highly trustworthy. Even on the rare occasions when the
security of our postal system does fail, it has mechanisms to quickly detect and to correct this.

Please replace the paragraph beginning at page 4, line 10, with the following rewritten paragraph:

15
a²
Accordingly, prior art cryptosystems and PKI systems provide many benefits, but even
they are not perfect in all regards. It is increasingly becoming apparent that it is now desirable to
improve on, augment, or even replace such systems.

20
Please replace the paragraph beginning at page 16, line 19, with the following rewritten paragraph:

a³
The text in the tables of FIG. 6a-d describes some of the particular fields, with the
primary fields discussed further presently. FIG. 6a is the users table 102 of FIG. 5. This contains
25 data records for each user, sender 12 or receiver 16, which is registered with the secure e-mail
system 10. As each user registers, they are assigned a UserId (userId 102a) and they choose a
Password (password 102b) which are stored here. The preferred value of the Password (password
102b) is $H(p + s)$ where p is the cleartext password and s is a salt (salt 102c) concatenated with
the cleartext password. FIG. 6b is the sentMail table 104 of FIG. 5. This contains data records for
30 each secure e-mail 14 in the secure e-mail system 10. FIG. 6c is the receivers table 106 of FIG.
5. This contains destination data for each secure e-mail 14 which is to be deliverable by the

a3
secure e-mail system 10. Since a record gets generated in this table for each receiver 16 (individual or list group) of each secure e-mail 14 that is sent, it is expected that this table will be the largest by far in the secure e-mail system 10. A null value in the FirstRequest field (firstRequest 106c) implies that the receiver 16 has not requested to read the secure e-mail 14.

5 FIG. 6d is the user aliases table 103 of FIG. 5. This contains data for all known email addresses (emailAddress 103a) for each given user (userId 103b, relationally linked to userId 102a in the users table 102). Thus single users may be known by multiple email addresses, or aliases.

10 **Please replace the paragraph beginning at page 19, line 18, with the following rewritten paragraph:**

a4
15 Here the random element is an anti-cracking feature, it is a large random number used to ensure that even e-mails that are the same in content are not the same when secured; the length element is the number of characters in the body field 60; the mic element is a message integrity code created by taking a hash of the body field 60; the subject element is the contents of the subject field 58; and the body element is the contents of the body field 60.

20 **Please replace the paragraph beginning at page 20, line 5, with the following rewritten paragraph:**

a5
25 1) If the emailAddress 103a for the sender 12 is unknown the encryption process 120 can determine a known emailAddress 103a or stop. The emailAddress 103a might be unknown for various reasons. One common example will be that the sender 12 is new to the security server 24. In this case the software module 26 can be directed to open a separate browsing window which allows the sender 12 to register on the spot. Another reason that the emailAddress 103a can be unknown is due to a user error. One simple source of such errors can be that multiple users share the same browser. A sender 12 can then be requested to clarify their identity.

30 **Please replace the paragraph beginning at page 24, line 20, with the following rewritten paragraph:**

a⁶/

5 In a step 166 the software module 26 validates the secure e-mail 14. This involves a second round of communications with the security server 24. The software module 26 generates new hashes of each part of the secure e-mail 14 and sends these and the seals included in each message part to the security server 24. The security server 24 then computes new seals, based on the passed in hashes, which it compares with the passed in seals. If there are any differences, this is an indication that the secure e-mail 14 is not authentic. The security server 24 then sends an indication about the authenticity of the secure e-mail 14 back to the software module 26.

10 **Please replace the paragraph beginning at page 25, line 15, with the following rewritten paragraph:**

a⁷/

15 The security features underlying the preceding encryption process 120 and decryption process 150 bear some further analysis. For authentication purposes, the operator of the security server 24 knows the sender 12 because their emailAddress 103a should associate with their password 102b. If the password 102b is treated the way it is supposed to be, i.e., only the holder should know it, then the operator of the security server 24 can be sure that only the sender 12 could have sent a particular secure e-mail 14. But the sender 12 does not necessarily even have to be trusted. By storing the sealSalt 104h initially, it is also possible for the operator of the security server 24 to be sure that no one, including the sender 12, can alter a secure e-mail 14 after it is sent. As an added security feature the sealSalt 104h may be stored encrypted in the database 100, and then never shared and never allowed to leave the security server 24. By encrypting the hashes of the body and attachments (H(b), H(a)) with the SSL key after the sender 12 has been authenticated (by providing the password 102b) it is possible to determine that it is the sender 12 who is signing their secure e-mail 14. Because the security server 24 stores only a hash of the actual password of the sender 12 as the password 102b, there is no way even the operator of the security server 24 can falsely sign a secure e-mail 14 on behalf of the sender 12.

30 **Please replace the paragraph beginning at page 28, line 1, with the following rewritten paragraph:**

a⁸/

The present secure e-mail system 10 is well suited for application in current network